HP OpenView Configuration Management

Configuration Server

for the AIX, HP-UX, Linux, and Solaris operating systems

Software Version: 5.00

Configuration Server and Database Migration Guide

Legal Notices

Documentation Updates

This guide's title page contains the following identifying information:

- Version number, which indicates the software version
- Document release date, which changes each time the document is updated
- Software release date, which indicates the release date of this version of the software

To check for recent updates or to verify that you are using the most recent edition, visit the following URL:

http://ovweb.external.hp.com/lpe/doc_serv/

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Support

You can visit the HP Software support web site at:

www.hp.com/go/hpsoftwaresupport

This Web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require a support contract.

To find more information about access levels, go to:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Contents

1 Upgrading the CM CS Database

This guide will help you upgrade an existing CM Configuration Server (CM-CS) and CM CS Database to version 5.00.

> Be sure to back up your existing database before beginning the upgrade process.

The upgrade process consists of two main steps:

1 Installing the Version 5.00 CM Configuration Server

2 Upgrading the existing CM CS Database to version 5.00. Version 5.00 CM CS Database is in UTF-8 format.

Follow the steps in this guide to successfully upgrade your CM Configuration Server and CM CS Database.

Installing the CM Configuration Server

Use the version 5.00 media to install the CM-CS. During the installation:

- Select the optional components CM OS Manager, CM Patch Manager and CM Usage Manager.

- Do not overwrite the existing CM-CS installation.

After you've installed the 5.00 CM-CS, remove the 5.00 CM CS Database, as this is not necessary for the migration process.

To remove the 5.00 CM CS Database

1 Stop the CM CS service.

2 Remove the DB folder, installed by default: /opt/HP/CM/ConfigurationServer/DB

When you've finished installing the CM-CS, update the new CM CS Database.

Upgrading the CM CS Database

Before you upgrade your HP OpenView Configuration Management Configuration Server Database (CM CS Database), it is strongly recommended that you use the Verify Database Utility— a non-invasive database analysis tool—to check the status and integrity of the elements in the Database. In general, it is recommended that you periodically verify the status of your Database and perform any necessary maintenance or cleanup.

Once your existing Database has been verified and you have performed any recommended maintenance, use the Database Difference Utility to compare your existing CM CS Database class templates to the new CM CS Database class templates. This utility creates a file that contains the verbs that will be used to update your Database. You should then review this file and specify what changes to include or exclude. When finished reviewing the file,

complete the several procedures which are included in this chapter to create the new Database.

Pre-requisites for the CM CS Database Upgrade

- CM Configuration Server 5.00 or above

System Requirements for the CM CS Database

- You must have free space that is equal to three times the size of your CM CS Database. For space limitation considerations, see Space Limitation Considerations on page 17.

Platform Support

- See the release note document that accompanies this release for detailed platform support

> The locations and directory names presented below are only suggestions used to help with the database upgrade procedure.

Using the Verify Database Utility

When using the Verify Database Utility to check the status and integrity of the elements in the database, you will:

- Create a backup copy of your current CM CS Database.

- Run the Verify Database Utility to verify the contents of your database.

- Review the resulting log and perform the recommended maintenance.

To use the Verify Database Utility

1 Stop the CM Configuration Server service.

2 Create a new directory, such as /mydatabase, that you will use to store a backup of your database.

3 Copy your existing CM Configuration Server database directory to the newly created /mydatabase directory. The default database location for previous CM CS installations was:

    /opt/Novadigm/ConfigurationServer/DB

4 Create an additional directory within the mydatabase directory to store the Verify Database Utility, such as /exe.

    Your directory structure should appear similar to the following:

    /opt
      |-/mydatabase
           |-/DB
           |-/exe

5 Create a backup copy of your EDMPROF file (.edmprof.orig, for example).

6   Extract the contents of the Verify Database Utility zip file found on the CM DVD in the `\management_infrastructure\configuration_server\migrate_db` directory to the to the `/exe` directory. The Verify Database Utility varies by operating system:

— `hpo_ams_vrfydb.tar` for HP-UX

— `sol_ams_vrfydb.tar` for Solaris

— `linux_ams_vrfydb.tar` for Linux

— `aix_ams_vrfydb.tar` for AIX

7   Verify permissions before continuing.

This tar file contains:

— ZEDMAMS

— five library files

8   Use a text editor and edit the `EDMPROF` file.

9   Go to the [MGR_DIRECTORIES] section and use the DBPATH argument to specify the path to the backup of your database.

```
[MGR_DIRECTORIES]
 DBPATH = /opt/mydatabase/DB
```

10 Save and close the file.

11 Change to the directory where you stored the Verify Database Utility (such as `/opt/mydatabase/exe/`).

12 To ensure the BEHAVIOR class follows standard naming conventions, run the following command against the database you will be migrating (such as `/opt/mydatabase/DB`):

```
./ZEDMAMS VERB=EDIT_CLASS_PREFIX,FILE=PRIMARY,DOMAIN=SOFTWARE,CLASS=
BEHAVIOR,PREVIEW=NO,KEEPDATE=YES,FIELD=CLASTYPE,VALUE=T
```

13 Rename `zedmams.log` to `zedmams_edit_class.log`.

14 Run the Verify Database Utility. The following is a sample command that you might use to verify the integrity of all of the domains in the database:

```
ZEDMAMS VERB=VERIFY_DATABASE,DOMAIN=ALL,LOGFILE=verifydb.log,DEPTH=
RESOURCE,ZFILE=verifyzfile.sh
```

15 Save a copy of `verifydb.log`.

Table 1   Parameters for Verify Database Utility

Parameter	Explanation
VERB	Specifies the verb that you are running. In this case, you must use VERIFY_DATABASE.
DOMAIN	Specifies the domain that you want to verify. DOMAIN=ALL Verifies all domains, except for the PROFILE domain, which cannot be verified using this utility. DOMAIN=*DomainName* Verifies the domain that you specify, such as SOFTWARE.

Parameter	Explanation
LOGFILE	An optional parameter that can be used to specify the name of the log that contains information about the status of your database. By default, the log file is `ZEDMAMS.log`. This log is stored in the same directory as the Verify Database Utility.
DEPTH	Use the depth parameter to restrict the verification in the database. The following are acceptable values: • resource • instance • class • domain Usage: **depth=**`(Domain/class/instance/resource)`
ZFILE	Use the zfile parameter to create a batch file with all of the verbs reported in the veify_database log to fix the inconsistencies in the database. Output can be executed using the following command. (Note: ZFILE is case sensitive): ZEDMAMS ZFILE (FILENAME) Type a fully qualified path and filename to designate the batch file.

> The Verify Database Utility may run for several minutes or several hours depending on the size of your Database.

16 Open the log (verifydb.log from the example above)) to review information about the state of your CM CS Database. For more information about any of the verbs in the recommended solutions see the HP Technical Support web site.

> While reviewing output, note that two classes SVR4DEP and SDDEP will be removed. Be sure to allow these classes to be removed from your database by leaving those lines uncommented.

17 Review `verifyzfile.sh`.

18 When satisfied, run the following command:

```
./ZEDMAMS ZFILE "verifyzfile.sh"
```

19 Save the log file (copy `zedmams.log` to `zedmams.ZFILE.log`).

20 After performing the recommended solutions, run the Verify Database Utility again and then review your `ZEDMAMS.LOG` file to ensure that the issues have been resolved. Messages containing Improper Instance Name in the ZEDMAMS log file can be ignored. A sample message follows:

```
<18:02:04:308>   Improper Instance Name <SAMPLE_INSTALL_REX> in Component
Class <PRIMARY.SOFTWARE.BEHAVIOR>
```

Run the Verify Database Utility as many times as needed to assist you in determining the health of your database.

If you continue to package and publish to your production Database, be sure to create another backup copy and run the Verify Database Utility again immediately before performing the upgrade. This ensures that all potential issues are analyzed.

After you start the CM Configuration Server with the migrated database, the Configuration Server log file may contain many duplicate object IDs within the Base Instance. These messages can be ignored. An example message follows:

```
ERROR, DUPLICATE OBJECT ID <DABCB40992AB> FOUND WHILE ADDING
<PRIMARY.SOFTWARE.BEHAVIOR._BASE_INSTANCE_>
```

Using the Database Difference Utility

Use the Database Difference Utility to compare your existing CM CS Database class templates to the target CM CS Database class template. You will need to do the following:

- Create export decks of your existing CM CS Database.

- Run the Database Difference Utility to compare your export decks (created from your backup Database) against the new 5.00 CM CS Database class template schema (CORE.XPC).

- Review and edit the resulting file, which contains the verbs that will be used to update the target Database.

Creating Export Decks of Your Existing Database

Create the following export decks from your existing Database:

- **CM CS Database Classes (.XPC)**
  Used with the Database Difference Utility to compare your existing CM CS Database against the new Database. Later, this is the source class deck that is imported into the new Database.

- **CM CS Database Instances (.XPI)**
  This is the source instance deck that is imported into the new Database.

- **CM CS Database Resources (.XPR)**
  This is the source resource deck that is imported into the new Database.

Use the export decks as a:

- Base for your comparison against the CM CS Database.

- Back-up file of your current CM CS Database.

To create an export deck of your current CM CS Database

1 Create a new directory, such as /opt/ExpDeck.

2 Make sure the CM Configuration Server service is stopped.

3 Change to the directory where you stored the Verify Database Utility, which contains the latest ZEDMAMS, such as, /opt/mydatabase/exe.

4 Create three export decks—one deck for the classes, one deck for the resources and one deck for the instances— in the directory that you created.

Each time you run ZEDMAMS, a new log file is created (by default
`ZEDMAMS.log`). If you would like to save the individual log files from each
ZEDMAMS session, either rename or move the default log file after you run
ZEDMAMS, or use the ZEDMAMS LOGFILE parameter and specify a new log
file name.

For example, after creating an export deck for your classes, rename the
`ZEDMAMS.log` file to `zedmams.xpc.log`.

— For example, to create an export deck for your classes named `OLDEXPORT.XPC`, run
the following:

```
./ZEDMAMS VERB=EXPORT_CLASS,FILE=PRIMARY,PREVIEW=NO,OUTPUT=/opt
/ExpDeck/OLDEXPORT.XPC
```

Save the log file (copy `zedmams.log` to `zedmams.XPC.log`).

— For example, to create an export deck for your instances named `OLDEXPORT.XPI`,
run the following:

```
./ZEDMAMS VERB=EXPORT_INSTANCE,FILE=PRIMARY,PREVIEW=NO,OUTPUT=/opt
/ExpDeck/OLDEXPORT.XPI
```

Save the log file (copy `zedmams.log` to `zedmams.XPI.log`).

— For example, to create an export deck for your resources named `OLDEXPORT.XPR`,
run the following:

```
./ZEDMAMS VERB=EXPORT_RESOURCE,FILE=PRIMARY,PREVIEW=NO,OUTPUT=/opt
/ExpDeck/OLDEXPORT.XPR
```

Save the log file (copy `zedmams.log` to `zedmams.XPR.log`)

You may receive the warning "no resource found". These warnings are reported on
the instances, which have no resources (non data-bearing instances).

After each command is run, review the `ZEDMAMS.LOG` file (located in the directory where you
stored the Verify Database Utility, (such as `/opt/mydatabase/exe`) to ensure that there
were no errors, and that your export completed with RC 0 or RC 4.

It may take a long time to create export decks of your entire CM CS Database.

Replacing NULL Characters

In some instances, the newly created XPC files will contain NULL characters that must be
replaced in order to import the XPC later. To remove these characters:

• Obtain the **null2blank.rxx** script from HP

• Run the script against your newly created XPC file. For example, to replace NULL
characters in OLDEXPORT.XPC, you would run the following command:

```
radrexxw null2blank.rxx OLDEXPORT.XPC
```

Next, use the Database Difference Utility to compare your existing CM CS Database class
templates with the target CM CS Database class templates.

Comparing Databases

Use the Database Difference Utility (dbdiffc.tcl) to compare the two class export decks.

To compare the databases

1   Create a new directory such as /opt/mydatabase/DBDiff.

2   Copy the following upgrade files into this directory.

— Your .XPC file (such as OLDEXPORT.XPC)

— The HP .XPC file (CORE.XPC located on the CM DVD in the
    \management_infrastructure\configuration_server\migrate_db folder).

— Database Difference Utility (dbdiffc.tcl extracted from the UNIX operating
    specific tar file xxx_dbdiff.tar).

— nvdkit (extracted from the UNIX operating system-specific tar file
    xxx_dbdiff.tar)

3   Navigate to the directory that you created (/opt/mydatabase/DBDIFF).

4   Run the Database Difference Utility. The following is an example of the command line:

```
./nvdkit dbdiffc.tcl SOURCE_CLASS_DECK=OLDEXPORT.XPC,
TARGET_CLASS_DECK=CORE.XPC,BATCH_OUT=output.sh,DELETE=N
```

Table 2   Required arguments for the Database Difference Utility

Argument	
SOURCE_CLASS_DECK	The filename of the export deck that you created from your existing Database.
TARGET_CLASS_DECK	The filename of the HP export deck (CORE.XPC).
BATCH_OUT	The name of the file that contains a list of the verbs to be executed after you review the file. Several text files will be created in the same directory where this file is stored. These files represent the new classes that will be added to the Database.
DELETE	If a field or class exists in the source XPC file, but does not exist in the HP supplied target XPC file (CORE.XPC): Set DELETE=Y to delete the fields or classes. Set DELETE=N to keep the fields or classes.

5   Use a text editor to open the output file (specified in the BATCH_OUT argument, for
    example output.sh).

6   Review this file and determine what changes you would like to keep. If you want to
    exclude any changes, place an asterisk (*) in front of the verbs that you do not want to
    run. If you want to include changes that are currently excluded (denoted by an asterisk
    at the beginning of the line), remove the asterisk.

> While reviewing output, note that two classes SVR4DEP and SDDEP will be
> imported. In order to remove these classes, comment out the import class
> verbs.

> The NOVADIGM domain will be imported by default. If this is not required for
> your implementation, comment out the appropriate import class verbs.

Below is a table that defines some of the most common ZEDMAMS verbs that you will encounter when reviewing the output file.

Table 3   Common Verbs in the Output File

Verb	Description
ADD_FIELD	Adds a variable at the end of a template. This verb has a KEEPDATE= option which allows for an update of a variable without updating the ZOBJDATE/ZOBJTIME of the Base_Instance.
CHANGE_FIELDNAME	Changes variable names in the class template. This verb has a KEEPDATE= option which allows for an update of the class without updating the ZOBJDATE/ZOBJTIME instance.
CHANGE_FLD_VALUE	Changes a template's variable length, type Configuration Server, and client flags. This verb has a KEEPDATE= option which allows for the value to be updated without updating the ZOBJDATE/ZOBJTIME instance.
CHANGE_INS_FIELD	Changes one field in each instance of a class and verifies connects. This verb has a KEEPDATE= option which allows for the field to be updated without updating the ZOBJDATE/ZOBJTIME instance.
CHANGE_INST_DATA	Globally changes data in instance records by class. This verb has a KEEPDATE= option which allows for the instance to be updated without updating the ZOBJDATE/ZOBJTIME instance.
COPY_FIELD	Copies attribute data to a new attribute or to an existing attribute. This verb has a KEEPDATE= option which allows for the field to be updated without updating the ZOBJDATE/ZOBJTIME instance.
DELETE_FIELD	Deletes an attribute from a template. This value does not alter the ZOBJDATE/ZOBJTIME instance.
DELETE_CLASS	Deletes a class from the database.

7   Save and close the file.

Once you have determined the differences between the CM CS Database class templates, you are ready to update your target CM CS Database.

Creating a New Database Directory

In this section, you will create a new directory for your new database.

To create a new database directory

1   Navigate to your newly installed CM Configuration Server directory. The default location is:

/opt/HP/CM/ConfigurationServer/

2   Create a new directory, such as /opt /HP/CM/ConfigurationServer/New_DB.

3   Within this new directory, create two additional directories:

/opt/HP/CM/ConfigurationServer/New_DB/PRIMARY

/opt/HP/CM/ConfigurationServer/New_DB/RESOURCE

These directories must be named PRIMARY and RESOURCE and the names must be capitalized.

4   Backup your EDMPROF file.

5   Use a text editor to open EDMPROF file.

6   Find the argument DBPATH in the [MGR_DIRECTORIES] section and change the value to point to the new database directory that you created.

[MGR_DIRECTORIES]

DBPATH = /opt/HP/CM/ConfigurationServer/New_DB

7   Save and close the file.

8   Copy the updated EDMPROF file to the UpdateDB directory.

Importing Your Class Deck

In this section, you will import your target class deck (XPC) into the new database. This provides a starting point for your new Database.

To import the target deck into the Database

1   Change the current directory to the /mydatabase/exe directory you created earlier, and run the following command:

```
./ZEDMAMS VERB=IMPORT_CLASS,FILE=/opt/mydatabase/DBDiff/CORE.XPC,
PREVIEW=NO
```

Be sure to include the correct fully qualified path to CORE.XPC in the command line.

2   Review the ZEDMAMS.LOG to ensure that there were no errors, and that your import completed with RC 0 or RC 4.

3   Save the log file (copy zedmams.log to zedmams.ImportXPC.log).

> ⚠️ If the number of class connections in your existing database does not match the number of connections in CORE.XPC, to avoid breaking any existing connections you will need to manually adjust the connections within the newly imported database before you import your existing database classes.

Running the Database Difference File

In this section, you will run the Database Difference file that you created when you ran the Database Difference Utility.

▶ Be sure that you have reviewed this file thoroughly and commented out any verbs that you do not want to be executed.

▶ Also, be sure no CM Configuration Servers are running before you proceed with the following steps.

1   Go to the directory where your BATCH_OUT file is stored (such as `/opt/mydatabase/DBDiff/`).

    Remember, this directory will also contain several files that represent the new classes that will be added to the database.

2   Copy `output.sh` to the `/opt/mydatabase/exe` directory.

    > You do not need to copy the text files to the exe directory. The `output.sh` file contains a fully qualified path for these files.

3   Change directory to the `/opt/mydatabase/exe` and run the following:

    `./ZEDMAMS ZFILE "output.sh"`

    where *output.sh* is the name of the file that you specified in the BATCH_OUT parameter when you ran the Database Difference Utility. Note that ZFILE must be in all caps.

4   Review and save the ZEDMAMS.LOG to ensure that there were no errors, and that your import completed with RC 0 or RC 4 (copy `zedmams.log` to `zedmams.ImportZFILE.log`).

Importing the Instances and Resources to the Database

Finally, you must import the remaining source export decks that you created earlier, as well as the new supplied decks (CORE.XPI and CORE.XPR) to finish building the database.

> Be sure no CM Configuration Server is running before you proceed with the following steps.

1   Copy the `.XPI` and `.XPR` files you created earlier (`OLDEXPORT.XPI` and `OLDEXPORT.XPR` for example) from the `ExpDeck` directory to the `/mydatabase/exe` directory.

2   Also copy CORE.XPI and CORE.XPR from the CM 5.00 DVD in the `\management_infrastructure\configuration_server\migrate_db` folder to the `/mydatabase/exe` directory.

3   Change the directory to `/mydatabase/exe` and run the following:

    `./ZEDMAMS VERB=IMPORT_INSTANCE,FILE=CORE.XPI, XPR=CORE.XPR,`
    `TIME=OLD,PREVIEW=NO,DUPLICATES=MANAGE,CONTINUE=YES,REPLACE=YES`

    > Each time you run ZEDMAMS, a new log file is created (by default `ZEDMAMS.log`). If you would like to save the individual log files from each ZEDMAMS session, either rename or move the default log file after you run ZEDMAMS, or use the ZEDMAMS logfile parameter and specify a new log file name.

4   Run the following command to import the new version 5.00 instances and resources:

    `./ZEDMAMS VERB=IMPORT_INSTANCE,FILE=OLDEXPORT.XPI, XPR=OLDEXPORT.XPR,`
    `TIME=OLD,PREVIEW=NO,DUPLICATES=MANAGE,CONTINUE=YES,REPLACE=YES`

    The filename used in this example represents the name of the export deck that you created earlier.

5   Review and save the `ZEDMAMS.LOG` to ensure that there were no errors, and that your import completed with RC 0 or RC 4 (copy `zedmams.log` to `zedmams.ImportXPI.log`).

6   Run the Verify Database utility and make sure no errors are reported in the log file other than Improper Instance Name errors, which can be ignored).

7   Start the CM Configuration Server.

8   To confirm the upgrade completed successfully, perform an Agent Connect (for example, open the CM Application Self-service Manager catalog). If you receive the same results as you did prior to the upgrade, then the upgrade was successful.

> If you commented out the import class verbs for the NOVADIGM domain, the import instance will result in 504 errors (because the 5.00 CM CS Database does not contain the NOVADIGM domain).

Your CM CS Database upgrade is now complete.

> If you currently use Distributed Configuration Servers and have recently upgraded your CM CS Database, the first time you synchronize your servers you must use CRC differencing. The default synchronization method (Control Differencing) is not sufficient. For instructions on how to change the differencing method, refer to the section, Differencing Techniques, in the Release Notes and Addendum for EDM DMA v4.04, available on the HP Technical support Web site.

> The connection to the agent computer will fail if any object in storage is larger than 6k bytes. A CM Agent cannot receive a heap size larger than 6k bytes (or 247 attributes). This does note include the README field, since the README field is not included in the transfer.
>
> Use the CM CSDB Editor to view actual instance sizes. Note that the object size may grow if more variables are added to the object.
>
> **To avoid an agent connection failure during data transfer:**
> Turn on the STATS [MGR_TRACE] option to print the size of the persistent object. The Configuration Server reports the object-size when it is deleted. Monitor this for any object that may grow larger than the current limit of 6k bytes.
> Additionally, the VAR trace option will display variable sizes that are being added to an object.
>
> For more information, refer to the *CM Configuration Server Guide*.

CM OS Manager Updates

⚠ After you upgrade, any customizations made to `rombl.cfg` in the LSB package will be lost. Use the CSDB Editor to save the existing `rombl.cfg`-(right-click the file and use the **Edit this Component** shortcut menu option).

Space Limitation Considerations

This upgrade process requires space available up to three times the size of your existing CM CS Database. If free space is limited, the following considerations can be applied:

- Do not copy the original database to create export decks.

- Delete backup copy of the original database before updating the target database.

- Move existing database RESOURCE folder to the target database. This allows you to avoid creating and importing large XPR files.

  ▶ Before moving a RESOURCE folder, be sure there no extended ASCII characters in the domain or class names of the existing database.

CM 5.00 CORE Database Changes

- The 5.00 CM CS Database format is UTF-8.
- The NOVADIGM domain was removed in version 5.00.
- ZBITARCH attribute added to AUDIT.ZSERVICE and AUDIT.PACKAGE
- RPTCLASS and TIMEOUT attributes added to AUDIT.WBEM
- ZBITARCH attribute added to SOFTWARE.ZSERVICE and SOFTWARE.PACKAGE
- DIRPERMS attribute added to SOFTWARE.UNIXFILE.

2 Validating the CM CS Database

The following errors provide information about structural issues that may be present in your Database.

> Unmated instances are instances that exist with no connection to a resource.

Table 4   Domain Validation

ERROR MESSAGE Impact	Suggested Solution
VERIFY DOMAIN <> CANNOT VALIDATE DMA INSTANCE COUNT. **High** impact for DMA Customers. **Low** impact for non-DMA Customers.	Run: ZEDMAMS VERB=REFRESH_DMA,PREVIEW=NO
VERIFY DOMAIN < > DMA INSTANCE COUNT DOES NOT EQUAL ACTUAL INSTANCE COUNT. **High** impact for DMA Customers. **Low** impact for non-DMA Customers.	Run: ZEDMAMS VERB=REFRESH_DMA,PREVIEW=NO

Table 5   Class Validation

ERROR MESSAGE Impact	Suggested Solution
MISSING TEMPLATE IN CLASS < >. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > HAS MISSING _BASE_INSTANCE. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > HAS INVALID _BASE_INSTANCE. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
VERIFY CLASS < > CANNOT READ DMA CLASS CONTROL INFORMATION. (INVALID CLASS). **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > PREFIX SIZE LESS THAN EXPECTED. (INVALID CLASS). **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.

ERROR MESSAGE Impact	Suggested Solution
CLASS < > PREFIX CONTAINS INVALID DATES. **Low** impact. For internal use only.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > PREFIX SIZE < > LESS THAN EXPECTED. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
DATABASE VERSION MISMATCH FOR CLASS < > **High** impact. Corrupt Class.	Delete the class. Restore the class from the last known good template for that class.
CLASS TEMPLATE FIRST ENTRY NOT FOUND. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
CLASS TEMPLATE < > TEMPLATE FIRST ENTRY NAME < > DOES NOT MATCH CLASS NAME.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > TEMPLATE SIZE MISTMATCH. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.
CLASS < > HAS HEAP SIZE < > WHICH IS GREATER THAN SUPPORTED MAXIMUM. **High** impact for resolution process and integrity of the class.	Delete the class. Restore the class from the last known good template for that class.

Table 6   Class Validation

ERROR MESSAGE Impact	Suggested Solutions
CLASS < > HAS MISSING _NULL_INSTANCE. * **Low** impact.	Policy needs to be reviewed to see if the instance is used. Administrator can create a _NULL_ instance in the class if necessary.
CLASS < > HAS INVALID _NULL_INSTANCE.* **High** impact.	Delete the class. Restore the class from the last known good template for that class.
VERIFY CLASS < > INSTANCE COUNT IS ZERO. **High** impact. Affects the System Explorer and resolution process.	Run: ZEDMAMS VERB=REFRESH_DMA,PREVIEW=NO
VERIFY CLASS < > CANNOT VALIDATE DMA INSTANCE COUNT. **High** impact. Affects the System Explorer and resolution process.	Run: ZEDMAMS VERB=REFRESH_DMA,PREVIEW=NO

ERROR MESSAGE Impact	Suggested Solutions
VERIFY CLASS < > DMA INSTANCE COUNT DOES NOT EQUAL ACTUAL INSTANCE COUNT. **High** impact. Affects the System Explorer and resolution process.	Run: ZEDMAMS VERB=REFRESH_DMA,PREVIEW=NO
INSTANCE SIZE LESS THAN MINIMUM REQUIRED. **High** impact. Affects the System Explorer and resolution process.	Delete the class. Restore the class from the last known good template for that class.

Table 7   Instance Validation

ERROR MESSAGE Impact	Suggested Solution
INSTANCE SIZE DOES NOT MATCH CLASS TEMPLATE. **High** impact. The instance is corrupt.	Delete the class. Restore the class from the last known good template for that class.
ZRSCDATE NOT FOUND BUT RESOURCE FLAG SET IN CONTROL INFO. **No** impact**.**	Delete the class. Restore the class from the last known good template for that class.
WARNING: UNMATED INSTANCE < > DOES NOT HAVE CORRESPONDING DATA. **Medium** impact. Resource is missing for an instance	Determine the impact the resource has on your database. If the resource is non-critical, delete. If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.
RESOURCE < > SIZE < > IS INVALID FOR ASSOCIATED INSTANCE < >. **High** impact. Damaged Resource.	Determine the impact the resource has on your database. If the resource is non-critical, delete. If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.

ERROR MESSAGE Impact	Suggested Solution
RESOURCE < > ACTUAL SIZE < > NOT EQUAL TO SIZE < > IN ASSOCIATED INSTANCE. **High** impact. Damaged Resource.	Determine the impact the resource has on your database. • If the resource is non-critical, delete. • If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.
ERROR: OPENING RESOURCE < > FOR INSTNACE < >.	Delete the class. Restore the class from the last known good template for that class.
ERROR: UNABLE TO READ PREFIX FOR RESOURCE < > FOR ASSOCIATED INSTANCE < >.	Delete the class. Restore the class from the last known good template for that class.
RESOURCE < > ACTUAL SIZE < > DOES NOT MATCH EXPECTED SIZE < > FOR ASSOCAITED INSTANCE. **High** impact. Damaged Resource.	Determine the impact the resource has on your database. • If the resource is non-critical, delete. • If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.
RESOURCE < > PREFIX CLASS NAME < > DOES NOT MATCH ITS ASSOCIATED INSTANCE < >. **High** impact. Damaged Resource.	Determine the impact the resource has on your database. • If the resource is non-critical, delete. • If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.

ERROR MESSAGE Impact	Suggested Solution
RESOURCE < > PREFIX INSTANCE NAME < > DOES NOT MATCH ITS ASSOCIATED NAME < >. **High** impact. Damaged Resource.	Determine the impact the resource has on your database. • If the resource is non-critical, delete. • If the resource is critical, set the ZSERVICE.REMOVAL attribute to A and redeploy. (Setting the REMOVAL attribute to A abandons the resource, relinquishing CM control.) Note: Deleting a resource may have adverse affects on other CM components. Before deleting a damaged resource, make sure it will not affect any existing components.
RESOURCE < > COMPRESSED SIZE < > NOT EQUAL TO SPACE < > IN VARIABLE < > OF ASSOCIATED INSTANCE < >. **Medium** impact. ZCMPSIZE Variable may be compromised.	Delete the class. Restore the class from the last known good template for that class.
RESOURCE < > SIZE < > NOT EQUAL TO SIZE < > IN VARIABLE < > OF ASSOCIATED INSTANCE < >. **Medium** impact. ZRSCSIZE Variable may be compromised.	Delete the class. Restore the class from the last known good template for that class.
RESOURCE < > CRC < > NOT EQUAL TO CRC < > IN VARIABLE < > OF ASSOCIATED INSTANCE < >. **Medium** impact. ZRSCCRC Variable may be compromised.	Delete the class. Restore the class from the last known good template for that class.

Checking for Orphaned Resources

Orphaned resources are resources that exist with no parent instance. The following error provides information about orphaned resources.

Table 8   Orphaned Resource Check

ERROR MESSAGE Impact	Suggested Solution
FOUND ORPHANED RESOURCE – DATA DOES NOT HAVE A CORRESPONDING PARENT INSTANCE < >. THIS RESOURCE WAS PREVIOUSLY PARENTED BY < >. **Low** impact. This resource can be left OR you may choose to remove the unmated resource by using the ZEDMAMS VERB = DELETE_RESOURCE.	Run: ZEDMAMS VERB=DELETE_RESOURCE, PREVIEW=NO

Checking for Duplicate IDs

The following errors provide information about duplicate object IDs that may be present in your Database.

Table 9   Object ID Check

ERROR MESSAGE Impact	Suggested Solution
DUPLICATE OBJECT ID < > WITHIN <COMPONENT> CLASS < > BETWEEN INSTANCE < > AND < >. **High** impact. Possibility of corrupt data.	If either data is corrupted, use: ZEDMAMS VERB = DELETE_INSTANCE, PACKAGE=YES,PREVIEW=NO,DOMAIN=< >, CLASS= < >, INSTANCE =< >
DUPLICATE OBJECT ID < > IN INSTANCES < > AND < >.	If both are from PACKAGE Class: Duplicate Instance object ids in Package Class cannot be fixed. You will need to delete the package. If class is NOT PACKAGE: ZEDMAMS VERB= CHANGE_OBJECTID,FILE=PRIMARY,DOMAIN= <>,CLASS=<>,INSTANCE=<>,PREVIEW =NO
DUPLICATE OBJECT ID < > WITHIN < > CLASS < > BETWEEN INSTANCE < >.	If class is PACKAGE: Duplicate Instance object ids in Package Class cannot be fixed. You will need to delete the package. If class is NOT PACKAGE: ZEDMAMS VERB=CHANGE_OBJECTID,FILE=PRIMARY,DOMAIN= <>,CLASS=<>,INSTANCE=<>,PREVIEW =NO

Index